ESAIM: PROCEEDINGS AND SURVEYS, Vol. ?, 2015, 1-10
Editors: Will be set by the publisher

HYDROMORPHO: A COUPLED MODEL FOR UNSTEADY STOKES/EXNER EQUATIONS AND NUMERICAL RESULTS WITH FEEL++ LIBRARY [*, **]

Nora Aïssiouene[1], Tarik Amtout[2], Matthieu Brachet[3], Emmanuel Frenod[4], Romain Hild[5], Christophe Prud'homme[6], Antoine Rousseau[7] and Stephanie Salmon[8]

Abstract. We propose to couple the Exner equation with the Stokes equations to model the sediment transport for bed load in geophysical flow. This work is a preliminary in context study to directly model the hydrodynamic flow by the unsteady Stokes equation instead of the classical shallow water equation. We focus in this proceeding to the coupling applying fluid structure interaction approach to morphodynamical behavior. In other words, we follow the approach of fluid interaction models replacing the structure equation by the Exner equation. The aim of this work is to validate the procedure proposed. These equations are solved by finite element method using the library FEEL++ (REF).

Résumé. ...

Introduction

In hydrodynamic field, many studies have been done to understand and predict the dynamical of sediments at the bottom of flow which is a significant and complex process for many geophysical situations. Morphodynamics modelling is a broad subject whose principles can be found in several texts [23], [24]. We can distinguish two types of sediments transport, the suspended load and the bed load. In this proceeding, we focus on the bed load transport and its impact on the hydrodynamics. The difficulty remains in the necessity to couple the sediment transport models with the hydrodynamic models which can be also complex in function of the application, and then to develop a robust and stable numerical method.

On the one hand, the sediments transport is usually modeled by the classical Exner equation [25] and several laws of transport have been proposed (see [13] to have delails of some classical laws) by physical arguments or

[*] ...

[**] ...

[1] Inria - CEREMA - Sorbonne Universités, UPMC Univ Paris 06 - CNRS, UMR 7598, Laboratoire Jacques-Louis Lions

[2] Faculté des Sciences et Techniques Tanger, Maroc

[3] Institut Elie Cartan de Lorraine, Université de Lorraine, Site de Metz, Bât. A Ile du Saulcy, F-57045 Metz Cedex 1

[4] Université Bretagne-Sud, LMBA UMR 6205, Vannes

[5] Institut de Recherche Mathématique Avancée, Strasbourg

[6] Institut de Recherche Mathématique Avancée, Strasbourg

[7] ...

[8] ...

closure relations. On the other hand, models as shallow water equations are used to model the hydrodynamics, and recently in [7, 14] a model derived from the Navier-Stokes equations that has an energy balance.

Concerning the numerical methods that have been established for these models, the main numerical schemes are developed for the hyperbolic systems with source terms for the hydrodynamic flow (see REF). Therefore, finite volume scheme are applied for the shallow water system ([1, 2, 26, 27]). The problem lies in the coupling of the numerical schemes. Indeed, in the shallow water models, the topography is a source term and the Exner equation gives the evolution of the bottom in terms of the velocity of the fluid. Then, two strategies are distinguished, the splitting one and the non-splitting one (REF). The splitting methods are easier to implement but generate instabilities in specific situations (see [9]). On the contrary, more complicated models, for instance involving relaxation, have to be used to take into consideration the fully coupled model ([3,18]).

Notice that the shallow water model is based on a hydrostatic assumption. It is deduced from the Navier Stokes equations, neglecting the vertical acceleration. Many other free surface models have been developed to take into account a non hydrostatic pressure ([5, 6, 21, 28]). However, numerical methods are much more complicated than for the hydrostatic model. It is not in the scope of this study to consider this kind of models, but it could be interesting to compare the results obtained with a non-hydrostatic model.

For this study, in order to take into account the non hydrostatic contribution, we chose to work with the unsteady Stokes equations.

The approach of this study is to apply the fluid structure interaction theory (see [8, 16]) to the coupled problem Stokes-Exner. We consider the sediments transport as a moving structure and use Stokes equations to model the fluid flow. Then, we use a Arbitrary Lagrangian Eulerian (ALE) method (see [16]) to couple the hydrodynamic flow and the bottom equation and we give a variational framework of the final model in order to apply finite element method.

Notice that the choice of the a finite element method applied to the Navier-Stokes equations implies to have costly numerical simulation, that is why we use a finite element library Feel++ within which parallelism is embedded. This allows to implement the method using variational formulation with high performance computing.

The proceeding is organized as follow, the first part is devoted to the description of the the model of the fluid and the description of the model of the bottom. In a second part, it is explained why a method like the ALE is necessary to couple the models. The third part establishes a complete ALE formulation of the Stokes-Exner model. Then, a variational formulation is given with the different boundary conditions that we explore. Finally, some numerical results are presented to evaluate the model and the solution method.

1. The model

For this preliminary study, we consider the domain

$$\Omega(t) = \{(x, z) \in \mathbb{R}^2 \quad | \quad 0 \leq x \leq l, \quad b(x, t) \leq z \leq 1\} \tag{1}$$

where $l > 0$ is the cavity length and $b(x, t)$ is the bottom topography in x at time t. We denote the velocity $\mathbf{u}(x, z, t) = (v(x, z, t), w(x, z, t))^T$, and the pressure $p(x, z, t)$.

We also denote by $\Gamma(t) = \Gamma_{in}(t) \cup \Gamma_{out}(t) \cup \Gamma_s \cup \Gamma_b(t)$ the boundaries (see Figure 1):

- $\Gamma_{in}(t) = \{0\} \times [b(0, t), 1]$
- $\Gamma_{out}(t) = \{l\} \times [b(1, t), 1]$
- $\Gamma_s = [0, l] \times \{1\}$
- $\Gamma_b(t) = \{(x, z) \in \mathbb{R}^2 \text{ s.t. } z = b(x, t), x \in [0, l]\}$

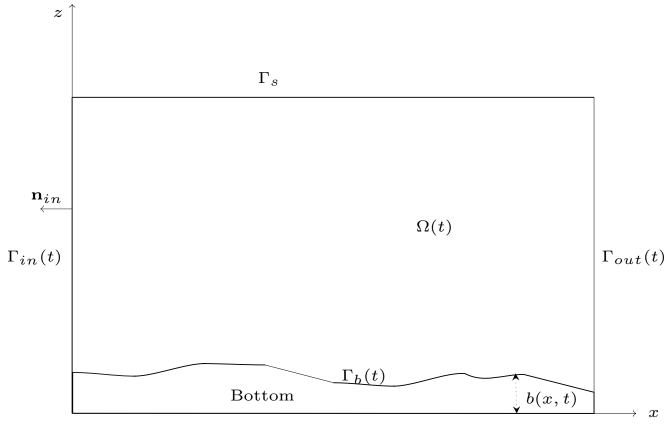


FIGURE 1. Definition of the domain

The coupled model leads to solving the non-steady Stokes problem in the fluid domain $\Omega(t)$ and the Exner equation to give the boundary $\Gamma_b(t)$.

1.1. Hydrodynamical Model

We consider the unsteady Stokes problem on the domain $\Omega(t)$

$$\frac{\partial \mathbf{u}}{\partial t} - \mu \triangle \mathbf{u} + \nabla p = 0 \text{ on } \Omega(t), \tag{2}$$

$$\operatorname{div}(\mathbf{u}) = 0 \text{ on } \Omega(t), \tag{3}$$

where $\mathbf{u} = (u, w)^T$ is the velocity of the fluid, p is the pressure and $\mu > 0$ is the viscosity. This problem is completed with the following boundary conditions.

- Dirichlet boundary condition :

$$\mathbf{u} = \mathbf{g} \text{ on } \Gamma(t) \tag{4}$$

In particular, we can look at the condition :

$$\mathbf{u} = 0 \qquad \text{on } \Gamma_{in}(t) \cup \Gamma_{out}(t) \cup \Gamma_b(t) \tag{5}$$

$$\mathbf{u} = (1,0)^T \text{ on } \Gamma_s \tag{6}$$

This is the case of the lid driven cavity.

- Mixed Dirichlet/Neumann/slip boundary condition :

$$\mathbf{u} = \mathbf{g}_1 \text{ on } \Gamma_s \tag{7}$$

$$p\mathbf{n} - \mu \frac{\partial \mathbf{u}}{\partial \mathbf{n}} = \mathbf{g}_2 \text{ on } \Gamma_{in}(t) \cup \Gamma_{out}(t) \tag{8}$$

$$\mathbf{u} \cdot \mathbf{n} = 0 \quad \text{on } \Gamma_b(t) \tag{9}$$

$$p\mathbf{n} - \mu \frac{\partial \mathbf{u}_\tau}{\partial \mathbf{n}} = \mathbf{g}_3 \text{ on } \Gamma_b(t) \tag{10}$$

The condition (7) impose the force by using Dirichlet condition.
The condition (8) let free the velocity at the inlet and at the outlet.
The condition (9) impose the normal component of the velocity to be null at the bottom. It is the

condition of impermeability of the domain.

The condition (10) let free the tangential component of the velocity at the bottom. It is needed to have a displacement of the bottom.

1.2. Morphodynamic model

The sediment dynamics is based on the formulation of a sediment continuity equation stating that the time variation of the sediment layer in a certain volume is due to the net variation of the solid transport through the boundaries of the volume. The mathematical expression of such law is known as the Exner equation [22] presented in this form :

$$\frac{\partial b_z}{\partial t} + \xi \frac{\partial Q}{\partial x} = 0 \text{ sur } [0, l] \tag{11}$$

where $b_z(x,t)$ is the bed elevation, $\xi = \frac{1}{1-p}$ is the material porosity and Q denote the solid transport discharge along the x coordinate influenced by the velocities u and w. The formulation of the bed load discharge Q can be based on deterministic laws ([4], [11], [29]) or in probabilistic methods ([12], [19]), often supported by experimentation. Grass [20] discussed one of the most basic sediment transport laws that in 1D can be written as

$$Q = \alpha |\mathbf{u}_\tau|^{3/2}$$

where $\alpha \in \mathbb{R}$ is an empirically parameter and represents the effects associated to the grain size and the kinematic viscosity. Ranging typically from 0 to 1, it represents a stronger interaction between flow and sediment as it approaches.

1.3. Arbitrary Lagrangian Eulerian Method (ALE)

In fluid mechanic, we can enumerate two ways to see the same problem : Lagrangian and Eulerian formulation.

Lagrangian formulation is similar to keep track the location of each fluid particles. The velocity $\mathbf{U}$ and the density ρ depend only of $\mathbf{x_0}$ the initial position of the particles and of t the time. Then, the time derivative of a quantity F is given by the total time derivative :

$$\frac{DF}{Dt}.$$

This method is interesting to understand or fixed the equation but it's costly to obtain numerical simulations.

For the Eulerian formulaton, the main idea is to fixed a system of coordinate and follow the flux of particles. In this context, the velocity $\mathbf{u}$ and the density ρ depend of $\mathbf{x}$ the position in a global system of coordinates and t the time. Now, for a function F, the time derivative is given by :

$$\frac{\partial F}{\partial t} + \mathbf{u} \cdot \nabla F$$

The link between the Eulerian and the Lagrangian time derivative is :

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla \tag{12}$$

In this works, we use Eulerian formulation for the fluid equation.

To solve numerically a problem, we must mesh the domain. Each point of this mesh have a coordinate but the domain is time dependant. It's saying that the mesh at time t is wrong at time $t + \Delta t$. Remesh at each time step is too expansive and keep only one mesh is impossible, so we want to move the mesh with the bottom.

This idea is classic in Fluid Structure Interraction : it's the Arbitrary Lagrangian Eulerian method. This method was first develloped for finite difference in [?, ?] and scope to finite element methods in [?, ?] and [?]. In 2004, [?] build a method for great order elements.

The main idea is to work in a reference domain $\widehat{\Omega}$ (in the following, we write $\widehat{\cdot}$ all quantities concerning the reference domain) to solve the fluid equation.

In our context, we choose $\widehat{\Omega}$ the rectangular $[0,l]\times[0,1]$ and the following boundaries :

- $\widehat{\Gamma}_{in} = \{0\}\times[0,1]$,
- $\widehat{\Gamma}_{s} = [0,l]\times\{1\}$,
- $\widehat{\Gamma}_{out} = \{l\}\times[0,1]$,
- $\widehat{\Gamma}_{b} = [0,l]\times\{0\} = \widehat{\gamma}_b\times\{0\}$ where $\widehat{\gamma}_b = [0,l]$.

To come back from the reference domain $\widehat{\Omega}$ to the physical domain $\Omega(t)$, the main tool is the ALE map :

$$\mathcal{A}^t : \left\{ \begin{array}{ccc} \widehat{\Omega} & \longrightarrow & \Omega(t) \\ \widehat{\mathbf{x}} & \longmapsto & \mathbf{x}(\widehat{\mathbf{x}},t) \end{array} \right. \tag{13}$$

We must determine $\mathcal{A}^t$ for every point of the mesh.

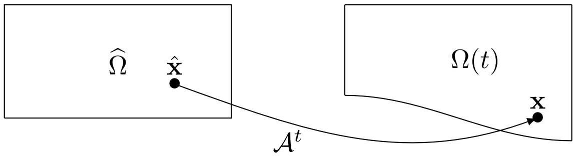


Figure 2. The ALE map

The point $\mathbf{x}(t)\in\Omega(t)$ can be found with :

$$\mathbf{x}(t) = \mathcal{A}^t(\widehat{\mathbf{x}}) = \widehat{\mathbf{x}} + \widehat{\mathbf{d}}_\delta(\widehat{\mathbf{x}},t) \tag{14}$$

where $\widehat{\mathbf{d}}_\delta(\widehat{\mathbf{x}},t)$ is the displacement of $\widehat{\mathbf{x}}$ between $\widehat{\Omega}$ and $\Omega(t)$. Notice that $\mathbf{x}(t)$ is time dependant. Then, we can define the mesh velocity :

$$\widehat{\mathbf{w}}(\widehat{\mathbf{x}},t) = \frac{\partial\mathcal{A}^t}{\partial t}(\widehat{\mathbf{x}}) = \frac{\partial\widehat{\mathbf{d}}_\delta}{\partial t}(\mathbf{x},t) \tag{15}$$

$\widehat{\mathbf{w}}$ is from $\widehat{\Omega}\times\mathbb{R}^+$ to $\mathbb{R}^d$. So, the equivalent from $\Omega(t)\times\mathbb{R}^+$ to $\mathbb{R}^d$ is $\mathbf{w} = \widehat{\mathbf{w}}\circ(\mathcal{A}^t)^{-1}$. We must determined the displacement $\widehat{\mathbf{d}}_\delta$ in $\widehat{\Omega}$. $\mathbf{d}_\delta$ must be smooth. In practice, $\mathbf{d}_\delta$ is the solution of a PDE like harmonic or Wislow equation. Sometimes, when the displacement is bigger, we can obtain a bad new mesh. Some methods are more effective than other to solve this (see [8] for more details). We often need to transport an equation from $\widehat{\Omega}$ to $\Omega(t)$ and mutually. Let $u:\Omega(t)\times\mathbb{R}^+\longrightarrow\mathbb{R}^d$, then the corresponding map in $\widehat{\Omega}$ is $\widehat{u} = u\circ\mathcal{A}^t$ If $\frac{\mathcal{D}\mathbf{u}}{\mathcal{D}t}$ is the times-derivative of u in ALE, we have the following equality :

$$\frac{\mathcal{D}\mathbf{u}}{\mathcal{D}t} = \left.\frac{\partial\mathbf{u}}{\partial t}\right|_x + \mathbf{w}\cdot\nabla\mathbf{u} \tag{16}$$

where $\dfrac{\partial\mathbf{u}}{\partial t}$ is the times partial derivative.

- If $\mathbf{w} = \mathbf{u}$, the mesh is moving with the particles so the description is Lagrangian.
- If $\mathbf{w} = \mathbf{0}$, the mesh don't move and the description is Eulerian.

1.4. **Coupled model**

In this part, we focus on the coupled model. We denote by $\widehat{\mathbf{x}}$ a point in the reference domain $\widehat{\Omega}$ and by $\Omega(t)$ the deformed domain after the transformation. The deformation of the mesh leads to consider the Lagrangian derivative $\mathcal{D}$ defined by (16) which is also called the ALE derivative where the velocity $\mathbf{w}$ is defined as: $\widehat{\mathbf{w}} = \frac{\partial \widehat{\mathbf{d}}_\delta}{\partial t}\Big|_{\hat{x}}$ and represents the velocity of the displacement of the mesh, that is to say the velocity of the particle in the referential domain . This allows to write the fluid model with a moving mesh $\Omega(t)$. Concerning the boundary conditions, we still consider a slip boundary condition at the interface between the topography and the fluid. The following model is still valid with the previous boundary conditions (4),(5)-(6),(7)-(10). The complete model is composed of the equation of the fluid in two dimensional domain, the equation of the topography in one dimensional domain and the ALE equation in two dimensional domain. After the changing variable (16), the coupled model reads:

1.4.1. *Fluid equation*

The fluid equation is given by :

$$\frac{\mathcal{D}\mathbf{u}}{\mathcal{D}t} - \mathbf{w}\cdot\nabla\mathbf{u} - \mu\triangle\mathbf{u} + \nabla p = 0 \quad \text{on } \Omega(t) \tag{17}$$

$$\operatorname{div}(\mathbf{u}) = 0 \quad \text{on } \Omega(t) \tag{18}$$

$$+\mathbf{BC} \tag{19}$$

1.4.2. *Bottom equation*

We consider the one dimensional domain $\widehat{\gamma}_b = [0,l]$, on which the bottom topography $\hat{b}(\hat{x},t) \quad \forall \hat{x} \in \gamma_b$ is defined by the Exner equation (11).

$$\frac{\partial \hat{b}_z(x,t)}{\partial t} + \frac{\partial \hat{Q}(\hat{x},t)}{\partial \hat{x}} = 0 \quad \forall \hat{x} \in \widehat{\gamma}_b, t > 0 \tag{20}$$

$$\hat{b}_z(\hat{x},0) = \hat{b}_{z,0}(\hat{x}) \tag{21}$$

Using EQREF, the sediment law $\hat{Q}$ is written in the reference domain by:

$$\begin{aligned}\hat{Q}(\hat{x},t) &= Q\left(\mathcal{A}^t\left((\hat{x},0)^T\right),t\right) \quad \forall \hat{x} \in \widehat{\gamma}_b &(22)\\ &= \alpha\left(u_\tau \circ \mathcal{A}^t\left((\hat{x},0)^T\right)\right)^{3/2} &(23)\end{aligned}$$

1.4.3. *Displacement equation*

This displacement needs to be extended in the fluid domain to associate a new ALE map over the mesh. In order to do this, we use a harmonic extension.

$$-\triangle\hat{\mathbf{d}}_\delta = 0 \quad \text{on } \hat{\Omega} \tag{24}$$

$$\hat{\mathbf{d}}_\delta = 0 \quad \text{on } \hat{\Gamma}_s \tag{25}$$

$$\frac{\partial \hat{\mathbf{d}}_\delta}{\partial \mathbf{n}} = 0 \quad \text{on } \hat{\Gamma}_{in} \cup \hat{\Gamma}_{out} \tag{26}$$

$$\hat{\mathbf{d}}_\delta = (0, \hat{b}_z(\hat{x},t))^T \quad \text{on } \hat{\Gamma}_b \tag{27}$$

This allows us to have a given displacement defined on $\hat{\Gamma}_b$, to let free the boundaries $\hat{\Gamma}_{in}$ and $\hat{\Gamma}_{out}$, and to fix the boundary $\hat{\Gamma}_s$. The harmonic problem diffuse the displacement $\hat{\mathbf{d}}_\delta$ on all the domain.

1.4.4. *Equation of w*

We denote by $\hat{\mathbf{w}}$ the velocity of the displacement

$$\hat{\mathbf{w}}(\widehat{\mathbf{x}},t) = \frac{\partial \hat{\mathbf{d}}(\hat{\mathbf{x}},t)}{\partial t} \tag{28}$$

Then, using the ALE transformation, we can compute the velocity $\mathbf{w}$ in the domain $\Omega(t)$:

$$\mathbf{w}(\mathbf{x},t) = \hat{\mathbf{w}}\left((\mathcal{A}^t)^{-1}(\mathbf{x}),t\right) \tag{29}$$

2. Variational formulation

This part is devoted in the variational formulation of the problem taking the ALE description into account. The problem leads to find $\mathbf{u} \in \mathbf{V}$ and $p \in Q$ such that the fluid equation (17) is satisfy. Let $\mathbf{X}$ be the functional set of test functions. Notice that the sets $\mathbf{V}$, $\mathbf{X}$ and Q will be define later. In practice, the sets $\mathbf{V}$ and $\mathbf{X}$ can be different, they depend on the boundaries conditions. By multiplying (17) with a test function $\mathbf{v} \in V$ and (18) by a test function $q \in Q$, and then integrating by part, we get :

$$\int_\Omega \frac{\mathcal{D}\mathbf{u}}{\mathcal{D}t}\cdot\mathbf{v} - \int_\Omega \mathbf{w}\cdot\nabla\mathbf{u}\cdot\mathbf{v} + \int_\Omega \nabla\mathbf{u} : \nabla\mathbf{v} - \int_\Gamma \frac{\partial\mathbf{u}}{\partial\mathbf{n}}\cdot\mathbf{v} - \int_\Omega p\,\mathrm{div}(\mathbf{v}) + \int_\Gamma p\mathbf{n}\cdot\mathbf{v} = \int_\Omega \mathbf{F}\cdot\mathbf{v} \tag{30}$$

$$\int_\Omega \mathrm{div}(\mathbf{u})\,q = 0 \tag{31}$$

Then, using the Reynolds transport formula on the first term of (30) :

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \mathbf{u}\cdot\mathbf{v} - \int_\Omega \mathbf{u}\nabla\cdot\mathbf{w}\cdot\mathbf{v} - \int_\Omega \mathbf{w}\cdot\nabla\mathbf{u}\cdot\mathbf{v}&\\ + \int_\Omega \nabla\mathbf{u} : \nabla\mathbf{v} - \int_\Gamma \frac{\partial\mathbf{u}}{\partial\mathbf{n}}\cdot\mathbf{v}&\\ - \int_\Omega p\,\mathrm{div}(\mathbf{v}) + \int_\Gamma p\mathbf{n}\cdot\mathbf{v} &= \int_\Omega \mathbf{F}\cdot\mathbf{v}\end{aligned} \tag{32}$$

$$\int_\Omega \mathrm{div}(\mathbf{u})\,q = 0 \tag{33}$$

We define the following forms :

$$\begin{aligned}
a_1(\mathbf{u},\mathbf{v}) &= \int_{\Omega(t)} \mu\nabla\mathbf{u} : \nabla\mathbf{v}\,dx & \forall\mathbf{u}\in\mathbf{V},\mathbf{v}\in\mathbf{V} && (34)\\
a_2(\mathbf{u},\mathbf{v}) &= -\int_\Omega \mathbf{u}(\nabla\cdot\mathbf{w})\cdot\mathbf{v} - \int_\Omega \mathbf{w}\cdot\nabla\mathbf{u}\cdot\mathbf{v}\,dx & \forall\mathbf{u}\in\mathbf{V},\mathbf{v}\in\mathbf{V} && (35)\\
a(\mathbf{u},\mathbf{v}) &= a_1(\mathbf{u},\mathbf{v}) + a_2(\mathbf{u},\mathbf{v}) & && (36)\\
b(\mathbf{u},q) &= \int_{\Omega(t)} q\,\mathrm{div}(\mathbf{u})\,dx & \forall\mathbf{u}\in\mathbf{V},q\in Q && (37)\\
L(\mathbf{v}) &= \int_{\Omega(t)} F(t)\cdot\mathbf{v}\,dx & \forall\mathbf{v}\in\mathbf{V} && (38)
\end{aligned}$$

3. Boundary conditions

In this part we explore various boundary conditions that are imposed to simulate for instance an inflow at the inlet, a free outflow at the outlet or a forcing flow on the top to represent the free surface.

3.1. **Dirichlet boundary conditions**

In many geophysical test, it is usual to simulate an inflow at the inlet of the domain by imposing a given velocity in the numerical method. In this part, we give the variational formulation corresponding to this situation and to simplify, we impose the Dirichlet condition for all the boundaries, which is not a physical situation. We introduce the spaces:

$$\mathbf{V} = \{\mathbf{u} \in \mathbf{H}^1(\Omega(t)), \quad \mathbf{u} = \mathbf{g} \text{ on } \Gamma\}$$
$$\boldsymbol{X} = \mathbf{H}^1_0(\Omega) = \{\mathbf{v} \in \mathbf{H}^1(\Omega(t), \quad \mathbf{v}|_\Gamma = 0\}$$

and

$$Q = L^2_0(\Omega(t) = \{q \in L^2(\Omega(t)), \quad \int_{\Omega(t)} q\, dx = 0\}$$

So, for the boundary conditions (4), the problem writes:
Find $\mathbf{u} \in \mathbf{V}$, $p \in Q$ such that

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \mathbf{u}\cdot\mathbf{v} - \int_\Omega \mathbf{w}\cdot\nabla\mathbf{u}\cdot\mathbf{v} - \int_\Omega \mathbf{u}\nabla\cdot\mathbf{w}\cdot\mathbf{v} + \int_\Omega \nabla\mathbf{u} : \nabla\mathbf{v} - \int_\Omega p\,\mathrm{div}(\mathbf{v}) = \int_\Omega \mathbf{F}\cdot\mathbf{v} \quad \forall\mathbf{v}\in\boldsymbol{X} \tag{39}$$
$$\int_\Omega \mathrm{div}(\mathbf{u})\, q = 0 \quad \forall q \in Q \tag{40}$$

With the notations (34)- (37), the problem writes :
Find $\mathbf{u} \in \mathbf{V}, p \in Q$ such that :

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \mathbf{u}\cdot\mathbf{v} + a(\mathbf{u},\mathbf{v}) + b\,(\mathbf{v},q) \;=\; L\,(\mathbf{v}) \qquad \forall \mathbf{v}\in\mathbf{X} \tag{41}$$
$$b(\mathbf{u},q) \;=\; 0 \qquad \forall q \in Q \tag{42}$$

3.2. Mixed Neumann/Dirichlet boundary conditions

In this part, we consider Dirichlet condition on the surface and we specify the boundary conditions that we impose on the inlet (resp. outlet) to simulate a free inflow (resp. outflow). This leads to impose a Neumann boundary conditions on the Stokes equations, which are the natural ones. Notice that in order to not treat the interface between the topography and the fluid in this part, we impose a Dirichlet boundary condition on the bottom boundary. We consider the problem (2)-(3) with the following boundary conditions:

$$\mathbf{u} \;= \mathbf{g}_1 \quad \text{on } \Gamma_s \cup \Gamma_b, \tag{43}$$
$$p\mathbf{n} - \mu\frac{\partial\mathbf{u}}{\partial\mathbf{n}} \;= \mathbf{g}_2 \quad \text{on } \Gamma_{in} \cup \Gamma_{out}. \tag{44}$$

and we define the spaces

$$\mathbf{V} = \{\mathbf{u} \in \mathbf{H}^1(\Omega(t)), \quad \mathbf{u} = g \text{ on } \Gamma_s \cup \Gamma_b\},$$
$$\mathbf{X} = \{\mathbf{u} \in \mathbf{H}^1(\Omega(t)), \quad \mathbf{u} = 0 \text{ on } \Gamma_s \cup \Gamma_b\},$$
$$Q = L^2_0(\Omega(t)).$$

Then the boundary terms in (30) writes

$$\int_{\Gamma_{in}\cup\Gamma_{out}} \left(p\mathbf{n} - \mu\frac{\partial\mathbf{u}}{\partial\mathbf{n}}\right)\cdot\mathbf{v} = \int_{\Gamma_{in}\cup\Gamma_{out}} \mathbf{g}_2\cdot\mathbf{v},$$

where g_2 is a given fonction. Then, the problem writes :
Find $\mathbf{u} \in \mathbf{V}$, $p \in Q$ such that:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \mathbf{u}\cdot\mathbf{v} + a(\mathbf{u},\boldsymbol{\varphi}) + b\left(\mathbf{v},q\right) \quad = \quad L\left(\mathbf{v}\right) \qquad \forall \mathbf{v}\in\mathbf{X} \tag{45}$$

$$b(\mathbf{u},q) \quad = \quad 0 \qquad \forall q \in Q \tag{46}$$

The bilinear form a and b are defined by (36)-(37), and L is defined by:

$$L(\mathbf{v}) \quad = \quad \int_{\Omega(t)} \mathbf{F}(t)\cdot\mathbf{v} \quad + \int_{\Gamma_{in}\cup\Gamma_{out}} \mathbf{g}_2\cdot\mathbf{v} \tag{47}$$

3.3. Slip boundary conditions

In this section, we are interested in the interface between the fluid and the topography and we preconise to have a slip boundary condition on Γ_b, which is physically consistent with the sediments transport model chosen in this study, namely the bed load transport.

Then, we give the variational formulation with slip boundary condition $\mathbf{u}\cdot\mathbf{n} = 0$ on Γ_b. To this aim,we rewrite the problem:
Find $\mathbf{u} \in \mathbf{V}, p \in Q$

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \mathbf{u}\cdot\mathbf{v} - \int_\Omega \mathbf{u}\nabla\cdot\mathbf{w}\cdot\mathbf{v} - \int_\Omega \mathbf{w}\cdot\nabla\mathbf{u}\cdot\mathbf{v} \\ + \int_\Omega \nabla\mathbf{u}:\nabla\mathbf{v} - \int_\Omega p\,\mathrm{div}(\mathbf{v}) + \int_\Gamma \left(p\boldsymbol{n} - \mu\frac{\partial\boldsymbol{u}}{\partial\boldsymbol{n}}\right)\cdot\mathbf{v} \quad = \int_\Omega \mathbf{F}\cdot\mathbf{v}\end{aligned} \quad \forall \mathbf{v}\in\boldsymbol{X} \tag{48}$$

$$\int_\Omega \mathrm{div}(\mathbf{u})\,q = 0 \;\forall q\in Q. \tag{49}$$

For the sake of clarity, we preconise a Dirichlet boundary conditions on the velocity for the boundaries $\Gamma_{in}, \Gamma_{out}$ and Γ_s, then the terms at boundaries vanish in the variational formulation and we focus on the on the interface Γ_b between the fluid and the topography. It is not natural to impose a slip boundary condition in the Stokes problem, and this problem has been widely studied to having a well posed problem with this boundary condition.

- A first strategy, studied in [10], consists in giving a condition on the stress tensor $\boldsymbol{\sigma\tau} = g$ defined by:

$$\boldsymbol{\sigma} = (\mu\nabla\mathbf{u} - p\mathbb{I}d) \tag{50}$$

  We rewrite the test function $\mathbf{v} = (\mathbf{v}\cdot\mathbf{n})\,\mathbf{n} + (\mathbf{v}\cdot\boldsymbol{\tau})\,\boldsymbol{\tau}$ where $\mathbf{n}$ is the normal component and $\boldsymbol{\tau}$ is the tangential component on Γ_b. Taking $g = 0$ and

$$\mathbf{v}\in\mathbf{V} = \{\mathbf{v}\in H^1(\Omega)^2\;,\;\mathbf{v}\cdot\mathbf{n}|_{\Gamma_b} = 0\}, \tag{51}$$

  it straightforward to verify that the variational formulation writes

$$\frac{\mathrm{d}}{dt}\int_\Omega \mathbf{u}\cdot\mathbf{v} - \int_\Omega \mathbf{u}\nabla\cdot\mathbf{w}\cdot\mathbf{v} - \int_\Omega \mathbf{w}\cdot\nabla\mathbf{u}\cdot\mathbf{v} + \int_\Omega \nabla\mathbf{u}:\nabla\mathbf{v} - \int_\Omega p\,\mathrm{div}(\mathbf{v}) \quad = \quad \int_\Omega \mathbf{F}\cdot\mathbf{v} \tag{52}$$

$$\int_\Omega \mathrm{div}(\mathbf{u})\,q \quad = \quad 0. \tag{53}$$

- A second strategy consist in giving a condition on the velocity at the boundary. To this aim, we notice that

$$\int_{\Gamma_b}\left(p\boldsymbol{n} - \mu\frac{\partial\boldsymbol{u}}{\partial\boldsymbol{n}}\right)\cdot\mathbf{v} \quad = \quad \int_{\Gamma_b}\boldsymbol{\sigma}\mathbf{n}(\mathbf{v}\cdot\mathbf{n}) - \int_{\Gamma_b}\mu\frac{\partial\boldsymbol{u}}{\partial\boldsymbol{n}}\cdot\boldsymbol{\tau}(\mathbf{v}\cdot\boldsymbol{\tau}) \tag{54}$$

We rewrite the test function $\mathbf{v}$ in terms of the normal component and the tangential component, as for the previous case. The problem writes:

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \mathbf{u}\cdot\mathbf{v} - \int_\Omega \mathbf{u}\nabla\cdot\mathbf{w}\cdot\mathbf{v} - \int_\Omega \mathbf{w}\cdot\nabla\mathbf{u}\cdot\mathbf{v}& \\
+\int_\Omega \nabla\mathbf{u}:\nabla\mathbf{v} - \int_\Omega p\,\mathrm{div}(\mathbf{v})& \\
+\int_{\Gamma_b} \sigma\mathbf{n}(\mathbf{v}\cdot\mathbf{n}) - \int_{\Gamma_b}\mu\frac{\partial\boldsymbol{u}}{\partial\boldsymbol{n}}\cdot\boldsymbol{\tau}(\mathbf{v}\cdot\boldsymbol{\tau}) &= \int_\Omega \mathbf{F}\cdot\mathbf{v}
\end{aligned}
\tag{55}
$$

$$
\int_\Omega \mathrm{div}(\mathbf{u})\,q = 0 \tag{56}
$$

Using the relation (54) and imposing $\frac{\partial\mathbf{u}}{\partial\mathbf{n}}\cdot\tau + \alpha(\mathbf{u}\cdot\boldsymbol{\tau})\boldsymbol{\tau} = g$ with $\alpha > 0$ (see [17]), the variational formulation writes:

$$
\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \mathbf{u}\cdot\mathbf{v} - \int_\Omega \mathbf{u}\nabla\cdot\mathbf{w}\cdot\mathbf{v} - \int_\Omega \mathbf{w}\cdot\nabla\mathbf{u}\cdot\mathbf{v} \tag{57}
$$

$$
+\int_\Omega \nabla\mathbf{u}:\nabla\mathbf{v} - \int_\Omega p\,\mathrm{div}(\mathbf{v})\int_\Gamma \alpha(\mathbf{u}\cdot\boldsymbol{\tau})(\mathbf{v}\cdot\boldsymbol{\tau}) \;=\; \int_\Omega \mathbf{F}\cdot\mathbf{v} + \int_\Gamma g(\mathbf{v}\cdot\boldsymbol{\tau}) \tag{58}
$$

$$
\int_\Omega \mathrm{div}(\mathbf{u})\,q \;=\; 0 \tag{59}
$$

- An other alternative leads to using a penalty method. As for the previous case, we take $\mathbf{v}\in\mathbf{V} = \{\mathbf{v}\in H^1(\Omega)^2\ ,\ \mathbf{v}\cdot\mathbf{n}|_{\Gamma_b} = 0\}$. In order to impose the condition $\mathbf{u}\cdot\mathbf{n} = 0$ for the velocity which is not natural in the variational formulation, we consider the formulation (48) and penalize the natural boundary condition :

  $$
  \left.\left(p\mathbf{n} - \mu\frac{\partial\mathbf{u}}{\partial\mathbf{n}}\right)\right|_{\Gamma_b} \;=\; \frac{1}{\epsilon}(\mathbf{u}\cdot\mathbf{n})\mathbf{n}
  $$

  where $\epsilon \ll 1$. The variational formulation becomes:
  Find $\mathbf{u}\in\mathbf{V}$ and $p\in Q$ such that:

  $$
  \frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \mathbf{u}\cdot\mathbf{v} + a(\mathbf{u},\mathbf{v}) + b(\mathbf{v},p) \;=\; L(\mathbf{v}) \quad \forall\mathbf{v}\in\mathbf{V} \tag{60}
  $$

  $$
  b(\mathbf{u},q) \;=\; 0 \quad \forall q\in Q \tag{61}
  $$

  with the bilinear form b defined by (37), L defined by (47) and a defined by

  $$
  a(\mathbf{u},\mathbf{v}) = a_1(\mathbf{u},\mathbf{v}) + a_2(\mathbf{u},\mathbf{v}) + \frac{1}{\epsilon}\int_{\Gamma_b}(\mathbf{u}\cdot\mathbf{n})(\mathbf{v}\cdot\mathbf{n})\,d\sigma \tag{62}
  $$

  It is proved by Dione in [10] that this problem converges to the problem with slip boundary conditions when ϵ tends to zero.

3.4. Variational formulation of the Exner equation

As for the fluid equation, a variational formulation of the the Exner equation is given for the complete method. Taking a test function $\phi\in L^2(\hat{\gamma}_b)$, it gives:

$$
\frac{\partial}{\partial t}\int_{\hat{\gamma}_b}\hat{b}_z(\hat{x},t)\phi(\hat{x})\,d\hat{x} + \int_{\hat{\gamma}_b}\frac{\partial\hat{Q}(\hat{x},t)}{\partial\hat{x}}\phi(\hat{x})\,d\hat{x} = 0 \tag{63}
$$

$$\frac{\partial}{\partial t}\int_{\hat{\gamma}_b} \hat{b}_z(\hat{x},t)\phi(\hat{x})\, d\hat{x} - \int_{\hat{\gamma}_b} \hat{Q}(\hat{x},t)\phi'(\hat{x})\, d\hat{x} + \left[\hat{Q}(\hat{x},t)\phi(\hat{x})\right]_{\hat{\gamma}_b} = 0 \quad \forall \phi \in L^2(\gamma_b) \tag{64}$$

The problem becomes : Find $\hat{b}_z$ such that for all $\phi \in L^2(\hat{\gamma}_b)$

$$\frac{\partial}{\partial t}\int_{\hat{\gamma}_b} \hat{b}_z(\hat{x},t)\phi(\hat{x})\, d\hat{x} = \int_{\hat{\gamma}_b} \hat{Q}(\hat{x},t)\phi'(\hat{x})\, d\hat{x} + \left[\hat{Q}(\hat{x},t)\phi(\hat{x})\right]_{\hat{\gamma}_b} \tag{65}$$

4. Numerical method

4.1. Discretisation in time

We will now discretise the time derivative with a Backward Differentiation Formula (BDF) of order 1. It correspond to the backward Euler method but we can consider a better method in a future research. Let Δt be the time step, $t_i = t_0$ the initial time, $t_n = n\Delta t$ and x^n a field at time t_n.
We can then rewrite the variational formulation at time t_{n+1} :

$$\int_\Omega \frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\Delta t}\cdot \boldsymbol{\varphi} + a(\mathbf{u}^{n+1},\boldsymbol{\varphi}) + b\left(\boldsymbol{\varphi}, p^{n+1}\right) = L\left(\boldsymbol{\varphi}\right) \quad \forall \boldsymbol{\varphi} \in \mathbf{V} \tag{66}$$

$$b(\mathbf{u}^{n+1}, q) = 0 \qquad \forall q \in Q \tag{67}$$

since $\mathbf{u}^n$ is known, this leads to :

$$\int_\Omega \frac{1}{\Delta t}\mathbf{u}^{n+1}\cdot \boldsymbol{\varphi} + a(\mathbf{u}^{n+1},\boldsymbol{\varphi}) + b\left(\boldsymbol{\varphi}, p^{n+1}\right) = L\left(\boldsymbol{\varphi}\right) + \int_\Omega \frac{1}{\Delta t}\mathbf{u}^{n}\cdot \boldsymbol{\varphi} \quad \forall \boldsymbol{\varphi} \in \mathbf{V} \tag{68}$$

$$b(\mathbf{u}^{n+1}, q) = 0 \qquad \forall q \in Q \tag{69}$$

4.2. Discretisation in space

We consider in this section a subdivided domain $\mathcal{T}_h$ and the finite dimensional spaces $\mathbf{V}_h$ and Q_h which are the discrete spaces of $\mathbf{V}$ and Q. As well, we approximate $\mathbf{u}$ and p by $\mathbf{u}_h$ and p_h. We use a Galerkin approximation taking the fields $\mathbf{u}_h$, and p_h in $\mathbf{V}_h$ and Q_h, it reads:

$$\mathbf{u}_h = \sum_{i=1}^{N} \alpha_i \boldsymbol{\varphi}_i, \quad p_h = \sum_{i=1}^{M} \beta_i \psi_i \tag{70}$$

where we denote $N = \dim \mathbf{V}_h$ and $M = \dim Q_h$ and $\boldsymbol{\varphi}$ (resp. ψ) is the basis function of $\mathbf{V}_h$ (resp. Q_h). Then, the semi discrete problem writes:

$$\int_\Omega \frac{1}{\Delta t}\mathbf{u}_h^{n+1}\cdot \boldsymbol{\varphi}_h + a(\mathbf{u}_h^{n+1},\boldsymbol{\varphi}_h) + b\left(\boldsymbol{\varphi}_h, p_h^{n+1}\right) = L\left(\boldsymbol{\varphi}_h\right) + \int_\Omega \frac{1}{\Delta t}\mathbf{u}_h^{n}\cdot \boldsymbol{\varphi}_h \quad \forall \boldsymbol{\varphi}_h \in \mathbf{V}_h \tag{71}$$

$$b(\mathbf{u}_h^{n+1}, q_h) = 0 \qquad \forall q_h \in Q_h \tag{72}$$

We choose the following compatible spaces (see [15]):

- $\mathbf{V}_h = \left\{\mathbf{v} \in \mathcal{C}^0\left(\overline{\Omega}\right) \text{ s.t. } \mathbf{v}_{|K} \in \mathbb{P}_2^2 \text{ for all } K \in \mathcal{T}_h\right\}$
- $Q_h = \left\{q \in \mathcal{C}^0\left(\overline{\Omega}\right) \text{ s.t. } q_{|K} \in \mathbb{P}_1 \text{ for all } K \in \mathcal{T}_h\right\} \cap L_0^2\left(\Omega\right)$

Where $\mathcal{T}_h$ is the set of mesh elements and $\mathbb{P}_k$ is the set of polynomial function of degree k. The space $\mathbf{V}_h$ can be adjust for specific boundary condition.
Throughout the rest of the document, we use this functionals sets.

We denote by U^n, P^n the vectors

$$U^n = \begin{pmatrix} \alpha_1^n \\ \vdots \\ \alpha_N^n \end{pmatrix}, \quad P^n = \begin{pmatrix} \beta_1^n \\ \vdots \\ \beta_M^n \end{pmatrix} \tag{73}$$

A the matrix $A_{i,j} = a(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_j)$ for $1 \leq i, j \leq N$ and B the divergence matrix $B^T_{i,j} = b(\varphi_i, \psi_j)$ for $1 \leq i \leq N, 1 \leq j \leq M$. We also note $F = (\gamma_i)^T$ where $\mathbf{f} = \sum_{i=1}^N \gamma_i \boldsymbol{\varphi}_i$ and M the mass matrix.
The problem writes:

$$\begin{pmatrix} A + M/\Delta t & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} U^{n+1} \\ P^{n+1} \end{pmatrix} = \begin{pmatrix} F \\ 0 \end{pmatrix} + \begin{pmatrix} M/\Delta t & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} U^n \\ 0 \end{pmatrix} \tag{74}$$

4.3. Stokes-Exner coupling

The algorithm for the Navier-Stokes equations constitute the main part of the complete method. The coupled algorithm can be summarized by the following:

Algorithm 1 Stokes-Exner coupling

```
Require: $\Omega^0$, $b^0$, $\mathbf{w}^0$, $\mathbf{u}^0$

  for $n = 0$ to $N_T - 1$ do
      $\mathbf{u}^{n+1} = \textbf{StokesSolver}(\Omega^n, \mathbf{u}^n, \mathbf{w}^n)$
      $b^{n+1} = \textbf{ExnerSolver}(b^n, \mathbf{u}^{n+1})$
      $\mathbf{d}^{n+1}_{\Gamma_b} = b^{n+1} - b^n$
      $\mathbf{d}^{n+1} = \textbf{ALESolver}(\Omega^n, \mathbf{d}^{n+1}_{\Gamma_b})$
      $\mathbf{w}^{n+1} = \frac{\mathbf{d}^{n+1} - \mathbf{d}^n}{\Delta t}$
      $\Omega^{n+1} = \Omega^n + \mathbf{d}^{n+1}$
  end for
```

5. Numerical tests

5.1. Analytical test for the Stokes problem

In order to validate our implementation of the Stokes problem, we use the known solution of Bercovier-Engelman (REF) :

$$\mathbf{v} = \begin{pmatrix} -256y(y-1)(2y-1)x^2(x-1)^2 \\ 256x(x-1)(2x-1)y^2(y-1)^2 \end{pmatrix}$$

$$p = (x - 0.5)(y - 0.5)$$

$$\mathbf{f} = \begin{pmatrix} 256(x^2(x-1)^2(12y-6) + y(y-1)(2y-1)(12x^2 - 12x + 2)) + (y - 0.5) \\ -256(y^2(y-1)^2(12x-6) + x(x-1)(2x-1)(12y^2 - 12y + 2)) + (x - 0.5) \end{pmatrix}$$

We can compare the exact solution and the approximation in Fig 3.

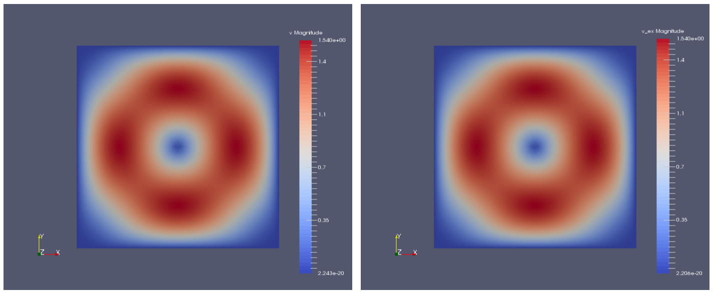
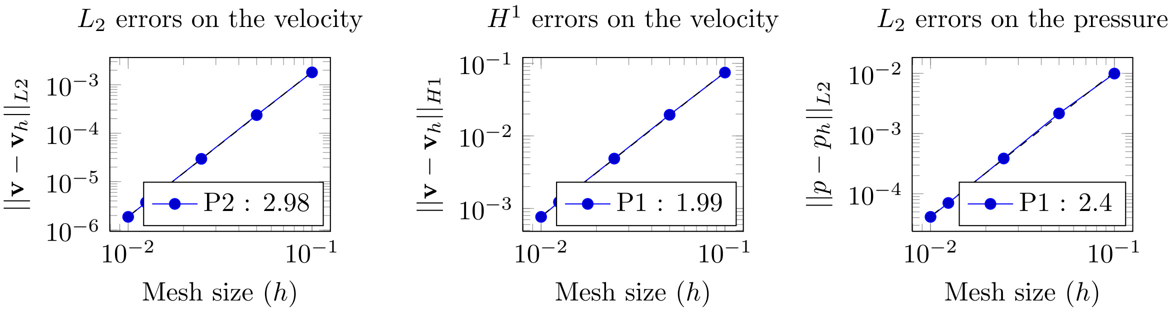


FIGURE 3. Magnitude of the velocity field. 3a Numerical, 3b Exact solution.



5.2. Driven cavity

We also study the boundary condition see in section **??**. For this, we want to reproduce a driven cavity. It means that we have $v|_{\Gamma_{in} \cup \Gamma_{out} \cup \Gamma_b} = 0$ and $v|_{\Gamma_s} = (1,0)^T$.
In Fig 4, you can see the expected behavior of the velocity.

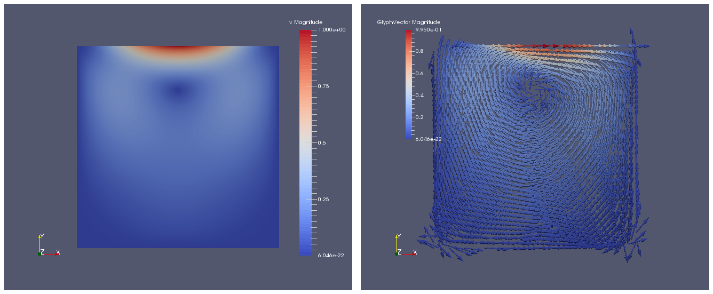


FIGURE 4. Velocity field. 4a Magnitude, 4b Vectors.

5.3. Fluvial dune test case

It is not straightforward to validate the method on the complete model with an analytical solution. It this part, the objective is to evaluate the coupling model with the classical test case of the dune migration. If we initialize the simulation with a succession of dunes and a fluvial flow, it is expected that the dunes migrate in the same direction as the flow.

6. Conclusion

In this note, we consider a coupling between the Exner equations and the Stokes equation to model the transport sediments in flow phenomena. We focus on a model without free surface and used some numerical test to evaluate the relevance of the method. The fluid structure interaction theory and method have been applied for the simplified and the objective is to test the proposed method which can be extend to a free surface model. The library Feel++ and the high computing performance embedded have been used to test the solution method. Therefore, the final goal of this project is to understand the impact of the sediment transport on the flow using Navier-Stokes with a free surface system coupled with the standard Exner equation.

7. Acknowledgments

A completer We acknowledge the GDR EGRIN for its financial support and the CIRM for its warm welcome during the CEMRACS.

8. Annexe: Implementation in Feel++ framework for the Stokes Problem

Feel++ is a C++ library for partial differential equation solves using generalized Galerkin methods such as the finite element method, the h/p finite element method, the spectral element method or the reduced basis method. The source code can be found here. We first need to define the type of the mesh that we'll use. For this, we need to precise the dimension and the type of elements used, simplex or hypercube. This is done with the lines :

```
typedef Simplex<2> convex_type;
typedef Mesh<convex_type> mesh_type;
typedef boost::shared_ptr<mesh_type> mesh_ptrtype;
```

The last line creates a pointer to a mesh.

Next, we have to precise the function space, from section **??** we know that we have to use Taylor-Hood function space. That is $\mathbb{P}_2^2 \times \mathbb{P}_1$.
We use the following lines to do so :

```
typedef Lagrange<2,Vectorial> base_u; // [P2]^2
typedef Lagrange<1> base_p; // P1
typedef FunctionSpace<mesh_type, bases<base_u, base_p> > space_type;
typedef boost::shared_ptr<space_type> space_ptrtype;
```

We now have the needed types, we can use them to define the problem.
First, we load the geometry by using the fonction `loadMesh`. This function takes several parameters, but most of them have a default value that we won't have to change.
In particular, the parameter `name` define which file to use, it is set by default to the value of the option `gmsh.filename`. Another mandatory parameter is the type of the mesh that we have defined previously.

```
auto mesh = loadMesh(_mesh=new mesh_type);
```

The keyword `auto` has been introduced in C++11, and permits to infer the type of a variable with the declaration. Since Feel++ has a intensive usage of the templates, the use of the `auto` keyword makes the code clearer. On the other hand, this hides the real type of the variables. Next, we declare the functoin space used. So we use the type previously defined. We can then get an element of this space by using the `element` method. At this point, we have an element $(\mathbf{v}, p) \in \mathbb{P}_2^2 \times \mathbb{P}_1$, and we can get each element of the tuple with the following lines :

```
auto VhQh = space_type::New( mesh );
auto VP = VhQh->element();
auto v = VP.element<0>();
auto p = VP.element<1>();
```

We can use the same method to define the test functions.

We're almost ready to define the variational formulation, but first we need to set the functions used as right hand side and boundary condition. We use the option to set them, this will allow us to change those functions at the execution, without the need to recompile the application.

```
auto f = expr<2,1>(soption("functions.f"));
auto g = expr<2,1>(soption("functions.g"));
```

What is happening here, is that we use the option `functions.f` and `functions.g` to define the functions used. The method `soption` transforms these options as a string (you have also the methods `doption`, `ioption` and `boption` to transform an option respectively as a double, an integer and a boolean).
The function `expr` will transform the string taken as parameter into an expression usable within Feel++. The templates parameters are the size of the expression, in this case, a matrix with 2 lines and 1 column, that is a 2D vector (in Feel++, the vectors are column vectors).
We can now define our variational formulation. We first set the linear form $\int_\Omega \mathbf{f} \cdot \boldsymbol{\varphi}$.

```
auto L = form1( _test=VhQh );
L = integrate(_range=elements(mesh),
              _expr=inner(f,id(phi)) );
```

The first line says that the lineat form operates on the composite space $\mathbb{P}_2^2 \times \mathbb{P}_1$. The second line, that L will be the integrale on all the elements of the mesh of the expression $\mathbf{f} \cdot \boldsymbol{\varphi}$.
The bilinear form operates on $(\mathbb{P}_2^2 \times \mathbb{P}_1) \times (\mathbb{P}_2^2 \times \mathbb{P}_1)$. Where one of them is the space where lives the test functions, and the other one is where lives the solution, it is called the **t**rial space.
The difference between these two spaces is made by the operators in the integrals. While the test functions have the usual operators, the trial ones use the operators with a suffix `t`. The list of the operators can be found here. Another useful suffix is `v`, it is the evaluation of a function.
We use the formulation described in section **??**, $\int_\Omega \nabla \mathbf{v} : \nabla \boldsymbol{\varphi} - p \nabla \cdot \boldsymbol{\varphi} - q \nabla \cdot \mathbf{v}$.

```
auto A = form2( _trial=VhQh, _test=VhQh);
A = integrate(_range=elements(mesh),
              _expr=inner(gradt(v),grad(phi)) );
A += integrate( _range=elements(mesh),
                _expr=-idt(p)*div(phi)
                     -id(q)*divt(v) );
```

Regarding the boundary conditions, we apply them strongly by the use of the `on` method of the bilinear form. This method takes as parameters the range on which apply the boundary condition, that is Γ. It takes also the right hand side associated with the bilinear form, and the element on which to apply those boundary condition. Finally, it takes the expression of the boundary condition.

```
A += on( _range=boundaryfaces(mesh), _rhs=L, _element=v, _expr=g );
```

We can change the range of the boundary, if we want to apply them only on Γ_{in}, we use `markedfaces(mesh, "in")` instead of `boundaryfaces(mesh)`. It is important however that the geometry has the marker "in" somewhere.

Finally, we can solve the system, simply by calling the method `solve` on the bilinear form, with as parameters, the right hand side and the element representing the solution.

```
A.solve(_rhs=L, _solution=VP)
```

The solver and preconditionner used to solve the system are implemented by PETSc. You can see some informations about them with repectively the `ksp-view` and `pc-view` options. You can then change what is used with the `ksp-type` and `pc-type` options.

References

[1] E Audusse. A multilayer Saint-Venant model˜: Derivation and numerical validation. *Discret. Contin. Dyn. Syst. Ser. B*, 5(2):189–214, 2005.

[2] E Audusse and F Bouchut. Kinetic Entropy Inequality and Hydrostatic Reconstruction Scheme for the Saint-Venant. *journal??*, pages 1–21, 2000.

[3] Emmanuel Audusse, Christophe Chalons, Olivier Delestre, Nicole Goutal, Magali Jodeau, Jacques Sainte-Marie, Jan Giesselmann, and Georges Sadaka. Sediment transport modelling : relaxation schemes for Saint-Venant - Exner and three layer models. *ESAIM: Proceedings*, 38:78–98, 2012.

[4] M. Larson B. Camenen. A general formula for non-cohesive bed load sediment transport. *Estuarine Coastal Shelf Sci*, page 249–260, 2005.

[5] Marie-Odile Bristeau, Nicole Goutal, and Jacques Sainte-Marie. Numerical simulations of a non-hydrostatic shallow water model. *Comput. Fluids*, 47(1):51–64, 2011.

[6] Marie-Odile Bristeau, Anne Mangeney, Jacques Sainte-Marie, and Nicolas Seguin. An energy-consistent depth-averaged Euler system: Derivation and properties. *Discret. Contin. Dyn. Syst. - Ser. B*, 20(4):961–988, 2015.

[7] M. J Castro Diaz, E. D Fernandez-Nieto, C Pares, and a. M Ferreiro. Two-dimensional sediment transport models in shallow water equations. A second order finite volume approach on unstructured meshes. *Comput. Methods . . .*, pages 1–38, 2009.

[8] Vincent Chabannes. *Vers la simulation des écoulements sanguins*. PhD thesis, school ??, 2013.

[9] S. Cordier, M.H. Le, and T. Morales de Luna. Bedload transport in shallow water models: Why splitting (may) fail, how hyperbolicity (can) help. *Adv. Water Resour.*, 34(8):980–989, 2011.

[10] Ibrahima Dione. *Analyse théorique et numérique des conditions de glissement pour les fluides et les solides par la méthode de pénalisation*. PhD thesis, school ??, 2013.

[11] R. Mueller E. Meyer-Peter. Formulae for bed-load transport. *Report on the 2nd Meeting International Association Hydraulic Structure Research*, page 39–64, 1948.

[12] H.A. Einstein. The bed-load function for sediment transportation in open channel flows. *Tech. Rep. 1026,US Department of Agriculture, Technical Bulletin*, 1950.

[13] E. D. Fernández-Nieto, T. Morales de Luna, G. Narbona-Reina, and J. D. Zabsonré. Formal deduction of the Saint-Venant-Exner model including arbitrarily sloping sediment beds and associated energy. *journal??*, pages 1–44, 2015.

[14] E. D. Fernández-Nieto, T. Morales de Luna, G. Narbona-Reina, and J. D. Zabsonré. Formal deduction of the Saint-Venant-Exner model including arbitrarily sloping sediment beds and associated energy. *journal??*, pages 1–44, 2015.

[15] V Girault and PA Raviart. Finite element methods for Navier-Stokes equations: theory and algorithms. *NASA STI/Recon Technical Report A*, 5:vii+200, 1986.

[16] C Grandmont. *Analyse mathématique et numérique de problèmes d'interaction fluide-structure. Application a la modélisation de l'appareil respiratoire*. Habilitation à diriger des recherches, Université Pierre et Marie Curie - Paris VI, November 2009. Dans le cadre de l'ANR M3RS (ANR-08-JCJC-0013-01).

[17] J. L. Guermond and P. D. Minev. A new class of massively parallel direction splitting for the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 200(23-24):2083–2093, 2011.

[18] Putu Harry Gunawan and Xavier Lhébrard. Hydrostatic relaxation scheme for the 1D shallow water - Exner equations in bedload transport. *Comput. Fluids*, 121:44–50, 2015.

[19] A. Kalinske. Movement of sediment as bed load in rivers. *???*, page 615–620, 1947.

[20] A. Kalinske. Sediments transport by waves and currents. *SERC London Cent. Mar. Technol., Report No. FL29*, 1981.

[21] D Lannes and F Marche. A new class of fully nonlinear and weakly dispersive green-naghdi models for efficient 2 d simulations. *J. Comput. Phys.*, 282:238–268, 2014.

[22] P. Nielsen. Movement of sediment as bed load in rivers. *Cambridge University Press*, page 615–620, 1947.

[23] P. Nielsen. Coastal Bottom Boundary Layers and Sediment Transport. *Advanced Series on Ocean Engineering*, vol. 4, 1992.

[24] P. Nielsen. Erosion and Sedimentation. *Cambridge University Press*, 1998.

[25] Gary Parker, Chris Paola, and Suzanne Leclair. Probabilistic Exner Sediment Continuity Equation for Mixtures with No Active Layer, 2000.

[26] B. Perthame and C. Simeoni. A kinetic scheme for the Saint-Venant system with a source term. *Calcolo*, 38(4):201–231, 2001.

[27] Benoit Perthame and J.-F. Gerbeau. Derivation of viscous Saint-Venant system for laminar shallow water; Numerical validation. *Discret. Contin. Dyn. Syst. - Ser. B*, 1(1):89–102, 2001.

[28] Jacques Sainte-Marie and Marie-Odile Bristeau. Derivation of a non-hydrostatic shallow water model; Comparison with Saint-Venant and Boussinesq systems. *journal??*, 2008.

[29] G. Smart. Sediment transport formula for steep channels. *J. Hydraul*, 3:267–276, 1984.